

We claim:

5

10

20

25

30

1. A method of estimating the results of a database query, the method comprising:

collecting workload information related to the database;
tracing query patterns of queries in the workload to identify the usage of
tuples in the database during execution of the queries;

determining sample weights based on tuple usage; and performing a weighted sampling of the database based upon the sample weights.

- 2. The method of claim 1 wherein the weighting sampling is based on a probability of usage of tuples required in executing the workload.
- 15 3. The method of claim 2 and further comprising computing an aggregate over values in each sampled tuple.
 - 4. The method of claim 3 wherein the aggregate is computed by multiplying each value by the inverse of the probability with which corresponding tuples were sampled.
 - 5. The method of claim 1 wherein the weights are a function of the frequency of access of a tuple and the number of queries in the workload that access the tuple.
 - 6. The method of claim 1 wherein the tuple usage is stored on a page level.
 - 7. A machine readable medium having instructions for causing a machine to perform a method of estimating the results of a database query, the method comprising:

collecting workload information related to the database;

tracing query patterns of queries in the workload to identify the usage of tuples in the database during execution of the queries;

determining sample weights based on tuple usage; and performing a weighted sampling based upon tuple usage.

5

- 8. The machine readable medium of claim 7 wherein the weights are a function of the frequency of access of a tuple and the number of queries in the workload that access the tuple.
- 10 9. The method of claim 7 wherein the tuple usage is stored on a page level.
 - 10. A system that estimates the results of a database query, the method comprising:

a module that collects workload information related to the database;

a module that traces query patterns of queries in the workload to identify the usage of tuples in the database during execution of the queries;

a module that determines sample weights based on tuple usage; and a module that performs a weighted sampling of the database based upon the sample weights.

20

25

15

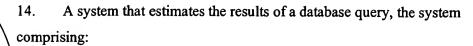
11. A method of estimating the results of a database query, the method comprising:

performing a weighted sampling of tuples in the database based on a probability of usage of tuples required in executing a given workload;

storing the probability for each tuple sampled;

computing an aggregate over values in each sampled tuple while multiplying by the inverse of the probability with which each tuple was sampled.

- 12. The method of claim 11 wherein the weights are a function of the frequency of access of a tuple and the number of queries in the workload that access the tuple.
 - 13. The method of claim 11 wherein the tuple usage is stored on a page level.



means for performing a weighted sampling of tuples in the database

based on a probability of usage of tuples required in executing a given workload;

means for storing the probability for each tuple sampled;

means for computing an aggregate over values in each sampled tuple while multiplying by the inverse of the probability with which each tuple was sampled.

10

15

25

30

15. A machine readable medium having instructions for causing a machine to perform a method of estimating the results of a database query, the method comprising:

performing a weighted sampling of tuples in the database based on a probability of usage of tuples required in executing a given workload;

storing the probability for each tuple sampled;

computing an aggregate over values in each sampled tuple while multiplying by the inverse of the probability with which each tuple was sampled.

20 16. A method of sampling tuples in a database to estimate the answer of an aggregation query, the method comprising:

determining which tuples are accessed more often during execution of a workload;

sampling the tuples; and

calculating an aggregate of values in the sampled tuples as a function of which tuples are accessed more often.

17. A method of generating an outlier index for a database and a given workload wherein the queries in the workload may have selection or group by conditions, the method comprising:

identifying sub-relations of tuples in the database induced by selection and group by conditions in queries in the workload;

generating a variance for values in each sub-relation;



25

selecting sub-relations having higher variances; and generating outliers from such sub-relations having higher variances.

- 18. The method of claim 17 and further comprising taking a union of outliers generated from such sub-relations.
 - 19. The method of claim 17 wherein sub-relations are selected having a variance higher than a desired threshold.
- 20. A machine readable medium having instructions for causing a machine to perform a method of generating an outlier index for a database and a given workload wherein the queries in the workload may have selection or group by conditions, the method comprising:

identifying sub-relations of tuples in the database induced by selection and group by conditions in queries in the workload;

generating a variance for values in each sub-relation; selecting sub-relations having higher variances; and generating outliers from such sub-relations having higher variances.

21. A system that generating an outlier index for a database and a given workload wherein the queries in the workload may have selection or group by conditions, the method comprising:

a module that identifies sub-relations of tuples in the database induced by selection and group by conditions in queries in the workload;

a module that generates a variance for values in each sub-relation;
a module that selects sub-relations having higher variances; and
a module that generates outliers from such sub-relations having higher variances.

30 22. A method of generating an outlier index for a database and a given workload wherein the queries in the workload may have aggregation and selection or group by conditions, the system comprising:

identifying sub-relations of tuples;



generating weights for each sub-relation based on workload information; generating a weighted variance for values in an aggregation column in each sub-relation;

allocating memory to sub-relations in proportion to their weighted variances; and

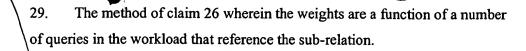
generating outliers from such sub-relations based on allocated memory.

- 23. The method of claim 22 and further comprising a module that takes a union of outliers generated from such sub-relations.
- 24. The system of claim 22 wherein sub-relations are selected having a weighted variance higher than a desired threshold.
- 25. The method of claim 22 wherein the weights are a function of a number of queries in the workload that reference the sub-relation.
 - 26. A method of generating an outlier index for a database and a given workload wherein the queries in the workload may have selection or group by conditions, the method comprising:

identifying sub-relations of tuples having values to be aggregated; generating weights for sub-relations based on workload information; generating a weighted variance for values in sub-relations; selecting sub-relations having higher weighted variances; and generating outliers from such sub-relations having higher weighted

- 25 variances.
 - 27. The method of claim 26 and further comprising taking a union of outliers generated from such sub-relations.
- 30 28. The method of claim 26 wherein sub-relations are selected having a weighted variance higher than a desired threshold.

10



A machine readable medium having instructions for causing a machine to perform a method of generating an outlier index for a database and a given workload wherein the queries in the workload may have selection or group by conditions, the method comprising:

identifying sub-relations of tuples having values to be aggregated; generating weights for sub-relations based on workload information; generating a weighted variance for values in the sub-relations; selecting sub-relations having higher weighted variances; and generating outliers from such sub-relations having higher weighted variances.

A system that generates an outlier index for a database and a given 31. 15 workload wherein the queries in the workload may have selection or group by conditions, the system comprising:

means for identifying sub-relations of tuples having values to be aggregated;

means for generating weights for each sub-relation based on workload 20 information;

means for generating a weighted variance for values in each sub-relation; means for selecting sub-relations having higher weighted variances; and means for generating outliers from such sub-relations having higher weighted variances.

32. A method of estimating the results of a database query over a relation, the method comprising:

defining weights of sub-relations using workload information; calculating a weighted variance for each sub-relation;

allocating memory to sub-relations in proportion to respective weighted variances;



5

10

25



building an outlier index for each sub-relation in accordance with allocated memory; and

taking the union of outlier indexes.

- 5 33. The method of claim 32 and further comprising retaining sub-relations having a weighted variance over a desired threshold prior to allocating memory to sub-relations.
- 34. The method of claim 32 wherein each weighted variance is a function of the weight of a sub-relation and the variance of the sub-relation.
 - 35. A method of estimating the results of a database and a given workload wherein the queries in the workload may have selection conditions, the method comprising:

collecting workload information related to the database;

tracing query patterns of queries in the workload to identify the usage of tuples in the database during execution of the queries;

determining sample weights based on tuple usage;
performing a weighted sampling of the database based upon the sample weights; and
generating a weighted outlier index.

- 36. The method of claim 35 and further comprising calculating an aggregate based on the samples and the index.
- 37. A method of estimating an aggregate result of a database and a given workload wherein the queries in the workload may have selection or group by conditions, the method comprising:

collecting workload information related to the database;

tracing query patterns of queries in the workload to identify the usage of tuples in the database during execution of the queries;

generating sample weights based on tuple usage; \
performing a weighted sampling based upon tuple usage;

25

30

15



5

15

20

30

generating an aggregate based on the weighted sampling;
identifying sub-relations of tuples in the database induced by selection and group by conditions in queries in the workload;

generating weights for each sub-relation based on workload information; generating a weighted variance for values in an aggregation column in each sub-relation;

allocating memory to sub-relations in proportion to their weighted variances;

generating outlier indexes from such sub-relations based on allocated nemory;

performing a union on the outlier indexes to form an outlier index for the relation;

computing an aggregate on the outlier index for the relation; and combining the aggregate based on the weighted sampling with the aggregate on the outlier index.

38. A method of estimating the results of a database and a given workload wherein the queries in the workload may have selection conditions, the method comprising:

building an outlier index on outlier values; building a sample of non-outlier values;

aggregating the outlier values and non-outlier values and scaling values as required.

- 25 39. The method of claim 38 wherein the outlier index is built using workload information.
 - 40. The method of claim 38 wherein the non-outlier sample is based on uniform sampling.
 - 41. The method of claim 38 wherein the non-outlier sample is based on weighted workload information.